RL-TR-94-178
In-House Report
November 1994

CONSTRUCTING A LEXICON FROM A MACHINE READABLE DICTIONARY

Michael L. McHale and John J. Crowter

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

Rome Laboratory
Air Force Materiel Command
Griffiss Air Force Base, New York

This report has been reviewed by the Rome Laboratory Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RL-TR-94-178 has been reviewed and is approved for publication.

APPROVED: Samuel A. DiNitto, Jr.

SAMUEL A. DINITTO, JR., Chief
Software Technology Division
Command, Control, & Communications Directorate

FOR THE COMMANDER: John A. Graniero

JOHN A. GRANIERO
Chief Scientist
Command, Control, & Communications Directorate

If your address has changed or if you wish to be removed from the Rome Laboratory mailing list, or if the addressee is no longer employed by your organization, please notify RL (C3CA) Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 1994	In-House

4. TITLE AND SUBTITLE
CONSTRUCTING A LEXICON FROM A MACHINE READABLE DICTIONARY

5. FUNDING NUMBERS
PE - 62702F
PR - 5581
TA - 27
WU - 67

6. AUTHOR(S)
Michael L. McHale, John J. Crowter

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Rome Laboratory (C3CA)
525 Brooks Road
Griffiss AFB NY 13441-4505

8. PERFORMING ORGANIZATION REPORT NUMBER
RL-TR-94-178

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Rome Laboratory (C3CA)
525 Brooks Road
Griffiss AFB NY 13441-4505

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
Rome Laboratory Project Engineer: Michael L. McHale/C3CA (315) 330-1458

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The production of Natural Language Processing systems that have both good syntactic coverage and broad lexical coverage has been the purview of large research teams. This effort initiated an investigation of an approach at empowering the smaller researcher with a method of incorporating a broad coverage, domain independent, syntactic parser with a large, general lexicon. The approach uses a Principle-Based Parser, based on Chomsky's Government-Binding Theory, and various on-line resources for lexical knowledge. This report outlines the tools, methods and results of the construction of the lexicon for the overall system.

14. SUBJECT TERMS
Artificial Intelligence, Natural Language Processing, Computational Lexicography NLP, MRD

15. NUMBER OF PAGES
52

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	U/L

Table of Contents

Chapter 1
Introduction

We dissect nature along lines laid down by our native language ... Language is not simply a reporting device for our experience but a framework for it.

B. Whorf

Abstract

The production of Natural Language Processing systems that have both good syntactic coverage and broad lexical coverage has been the purview of large research teams. This effort initiated an investigation of an approach at empowering the smaller researcher with a method of incorporating a broad coverage, domain independent, syntactic parser with a large, general lexicon. The approach uses a Principle-Based Parser, based on Chomsky's Government-Binding Theory, and various on-line resources for lexical knowledge. This report outlines the tools, methods and results of the construction of the lexicon for the overall system.

Background and Goal

Current Natural Language Processing (NLP) systems are doing an excellent job in helping users cope with databases, in understanding stylized messages, in translating technical documents and in retrieving pertinent information from financial databases. These are all important but constrained applications. A more productive use of NLP could be realized if the technology could be expanded to handle unconstrained language. Then the list of accomplishments could read: in helping users use computers, in understanding messages and texts, in translating newspapers and correspondence, and in retrieving all pertinent information across the World Wide Web. Some of these are wishlist items but in many cases the pressure to handle unconstrained text is already being felt. In military C^3I for instance, the amount of communications and information available to commanders has already risen to the point that commanders are unable to analyze all the pertinent information. Since much of this information is language based, NLP could assist in the analysis but only if the present NLP technology can be extended.

In order to realize such a comprehensive NLP system, syntactic parsers must be built that are robust, and have both broad coverage and efficiency. Current work in the computational linguistics field on parsing technology is geared toward making existing parsers more efficient. This emphasis on efficiency is required in part by the use of rule based grammars that have hundreds and even thousands of rules. With this type grammar there is a trade off between coverage and efficiency: the larger the domain, the larger the grammar with more corresponding rules. This results in less efficiency. Luckily there is an alternative.

Grammars based on Chomsky's Government-Binding Theory (Chomsky 1981, 1982, 1986) promise broad coverage of natural language syntax without the need for numerous phrase structure rules. The price for this simplified grammar is paid for partly by a rich lexicon, much richer than

that needed for a standard rule based grammar. This reliance on lexical knowledge has limited Principle Based Parsers[1] (PBP) to relatively small domains. Some of those researchers without the necessary resources for developing extensive lexicons have been investigating the characteristics of PBPs while those NLP researchers with large lexical resources have used more traditional grammars.

Previous NLP researchers (notably Heidorn, Jensen et al. of the IBM Epistle system) have used Machine Readable Dictionaries (MRD) though none of them have used a PBP. The reason for this became evident when preliminary examination of our base MRD indicated that only a portion of the information required by a PBP is explicitly represented. However, it was felt that the remaining information needed by the grammar is encoded implicitly in the dictionary. Much of the work in this effort, then, involved the design and testing of procedures to explicate this implicit information. The end result of explicating this information should allow using the MRD as the lexicon for a robust syntactic system at a fraction of the effort of what would be required if the lexicon were produced by hand.

The combination of Chomsky's Government-Binding Theory and an on-line lexicon promises to greatly ease the creation of broad coverage NLP systems for generalized text. Furthermore, if this combination is feasible, the resulting system would be much more portable and reusable than most of the NLP systems currently being used.

Original Approach

The approach originally developed for this effort was very simple and relied on proven NLP methodology — build it and see if it works. In this case that meant either writing or acquiring and modifying a PBP as well as

[1] The linguistic theory is known as Government-Binding theory. Both Government and Binding are principles of the theory but there are quite a few more. In order to make the name better reflect the composition of the theory some researchers renamed the theory Principles and Parameters and NLP systems that use this approach Principle Based Parsers.

extracting all the lexical information needed by the parser from on-line lexical sources.

The only PBP to which we had access was designed for Machine Translation (Dorr 1987). Modifying this parser seemed a formidable task but the parser did provide us with some early insight pertaining to the lexical needs of this type parser.

We were already somewhat familiar with two lexical resources: Longman's Dictionary of Contemporary English (LDOCE) and Roget's International Thesaurus, 3rd. edition (RIT). Our experience with LDOCE involved converting the on-line version into a readable lexicon and some other related work. LDOCE is somewhat unique in that the on-line version contains more information than the printed version. It was the extra information that enticed us into using LDOCE. It was felt that the subject field codes, grammar codes and other information would complement the contents of the printed version to the point that any information required by the parser could be deduced.

Our experience with RIT was mainly as computational users. The on-line version of RIT contains the same information as the printed version but only in one order - alphabetical. The printed version of RIT also contains a category order listing that makes the hierarchical nature of RIT somewhat more apparent. It is the hierarchical nature of RIT that we exploit in this research.

Our original goal and approach was to build a PBP and extract the lexicon from LDOCE supplementing it as necessary from RIT. To accomplish this, we figured, would require approximately 2.8 years of effort over the 24 month period of the research. The effort would involve writing the grammar and parser[2], extracting the lexicon, integrating the two parts, extensively testing the system and developing a demonstration model.

[2] It should be noted that the terms grammar and parser are not synonymous. The grammar being used here is GB, the actual parser will not be discussed but since the system is referred to as a PBP the term parser will be used to refer to the system.

Actual Approach

In actuality we fell far short of these goals. The actual time spent for the effort was 1.5 years of effort over the 24 month period. This disparity was caused by personnel reassignments and extra duties for remaining key personnel. This loss of research hours required extensive modification of our approach.

There were a number of options available to us for modifying the approach. The first was to do nothing, continue as planned and make the best of it when the time expired. This was not selected as the effort might have finished half way through the integration of the parsing system with the lexicon which would have left us with mostly unanswered questions.

The second option was to scale back on the lexicon and everything else except the parsing system. This would have allowed finishing the task in the allotted time but would not have met our goal of demonstrating the feasibility of producing a robust system with minimal assets. This option also would not have involved a great deal of original research in that a number of small PBPs have already been written.

The third option would be to completely replan the approach and goals to bring them into line with the new allotment of assets. This was selected as being the most viable of the options. We decided to postpone working on the parser and put all of our effort into the extraction and design of the lexicon. This allowed us to thoroughly investigate the lexicon in ways that we would not have been able to with the original plan. One result of this deeper investigation was the development of a method of using RIT to integrate disparate lexicons. This method will be covered in Chapter 5.

Overview

Chapter 2 discusses the design of the lexicon. Our study of Dorr's PBP, papers on various PBPs (see bibliography for listing) and communications with other researchers (Stabler, Berwick, Johnson) allowed us to determine the lexical requirements of a PBP. Using those requirements, along with our

overall design requirements of running the system as a parsing system and not as a browser or some other type of lexical resource, we were able to generate a design for the lexicon. Chapter 2 discusses both the physical and the logical design of the lexicon.

Chapter 3 covers the extraction of the extrinsic information found in the dictionary. It also discusses briefly some of the problems of going from a general ASCII based dictionary to a machine readable lexicon.

Chapter 4 covers the extraction of intrinsic knowledge from the dictionary, in particular, the extraction of thematic relations. This task was the main time consumer of all the tasks.

Chapter 5 discusses using RIT as a source of lexical enhancements. The chapter also discusses an algorithm developed for integrating RIT with other sources. Two spin-offs from this work are also covered. One was a patent application for the integration algorithm and the other was a multi-media (hypertext) demonstration of an integrated RIT/LDOCE lexical browser for aero-space terminology. Much work remains to be done in this area.

Chapter 6 discusses the results, summarizes and evaluates the overall research with suggestions for further investigations.

Chapter 2
Design of the Lexicon

Keep things as simple as possible,
but not simpler.

A. Einstein

Introduction

This chapter is divided into four main parts. The first part will give an overview of the minimal requirements of a PBP lexicon. The second part will discuss the physical structure that we are using for the lexicon. The third part will discuss *Longman's Dictionary of Contemporary English* (LDOCE) which is being used as the basis for our lexicon. The chapter will conclude with an outline of *Roget's International Thesaurus* and its relational hierarchy, which is being used both to enhance the semantics and to allow the integration of other dictionaries with the lexicon.

Requirements of a PBP Lexicon

Government-Binding is a lexical theory of grammar. Therefore, a PBP can be characterized as a lexical parser. This characterization implies that a PBP needs more information in the system's lexicon than some other parsing systems would require. For instance, a Definite Clause Grammar (Pereira and Warren 1980) can be written that only requires the words and their corresponding syntactic categories. In contrast to this simplicity, a PBP requires at a minimum the information given in Figure 2.1.

Information	Comments
word	the word and its various morphological forms
syntactic category	noun, verb, conjunction, etc.
subcategorization features	the types of syntactic complements that are associated with the word
thematic roles	AGENT, THEME, LOCATIVE, etc.
control information	subject-raising verb

Figure 2.1. Minimal lexical requirements for a PBP

Much of this information is explicitly represented in LDOCE in a variety of ways. Thematic roles (so called θ-roles) are a notable exception.

The Lexical Knowledge base

The organization of LDOCE is in the form of a lexical database designed for fast retrieval of individual words. It was decided that the lexical database should be in the form of a *trie* (Fredkin 1960). A trie (from the word re*trie*val) is a recursive tree structure that uses the characters of the word to direct the branching. Tries have traditionally been used for either the structure of a single disk file or for a file in memory. In this case the trie represents the directory structure and the letters of the word are the path name for the file. Each entry in the dictionary then is contained in the directory that has a path name being composed of the letters of the word. It was empirically determined that only the first seven letters of the word need be used (or fewer if the word has less than seven letters). The seven letter limit allows for the complete handling of LDOCE with no more than 37 files in any directory; well within the capacity of the operating system. Thus, the directory c:\c\o\n\t\e\n\t would contain the files for *content*, *contented*, *contention*, *contentment* and *contents*. The directory c:\l\o\v\e would contain the file *love* along with the sub-directories \r (*lover*), \i (*loving*), etc.

This structure allows us to handle a very large lexicon and shifts the burden of searching the knowledge base from the parser to the highly optimized operating system. It also eliminates the need for hash tables and the corresponding hashing functions and functions for conflict resolution. It does, however, limit the uses to which the knowledge base can be put. For instance, finding a particular word and its morphological forms is easy; a task that is constantly required by the parser. However, browsing the dictionary for semantically related words, or words with the same syntactic category or words related by anything other than orthography would be much more difficult.

Longman's Dictionary of Contemporary English

Longman's Dictionary of Contemporary English (LDOCE) provides the basis for the lexicon for the principle based parser being used for this research. LDOCE (1987) is designed for use by learners of English as a second language. It therefore demonstrates some differences from the ordinary English dictionary. A number of these differences are important to note.

The 55,000 or so words and phrases that are included in the dictionary were chosen for appropriateness as both core vocabulary and relevancy of current use. This choice of vocabulary results in the dictionary being a prime candidate for a basic NLP lexicon as there are fewer arcane or rare words than found in most dictionaries.

Additionally, the words are defined using a defining vocabulary of approximately 2000 basic words, thus the definitions are more easily understood. This limiting of the defining vocabulary, though, is a two-edged sword. On one side it cuts through the need to have a large vocabulary to understand the definitions. This would allow a computational system to "bootstrap" the definitions, that is, to work with the defining vocabulary and add other words as their definitions are processed. This approach would be very encouraging except that limiting the vocabulary increases the syntactic complexity of many of the definitions. This more complex syntax occurs especially in imbedded clauses that are in effect imbedded definitions. For instance, in the definition

computer an ELECTRONIC machine that can be supplied with a PROGRAM (= plan of operations) and can store and recall information, and perform various processes on it.

there is an imbedded definition of *program*.[3] This increase in syntactic complexity makes the processing of definitions more complicated than might be the case with other dictionaries.

Another feature of LDOCE is the inclusion of example sentences. LDOCE contains over 75,000 example sentences, many culled from American and British newspapers. These sentences are designed to provide natural and typical examples of each word's usage. For a learner of the language, these example sentences provide an aid for learning the correct way to use a word. The sentences can also play this role for NLP systems by providing grammatically correct sentences on which to test the parser. If all the parameters are correctly set and each principle in the parser is working correctly, then the sentences should parse. Failure to do so would indicate a problem with the parser. This testing also provides the opportunity to discover the correct number of arguments for the verbs, and the nature of the semantic roles filled by the nouns. For instance, if the enhanced lexicon indicates that a word should have a θ-role of LOCATIVE and the example sentence does not contain a LOCATIVE then the θ-role would require re-examination.

The above information is in both the hard-copy and on-line versions. The on-line version, though, has information that is not present in the hard copy version. For instance, the verb *saddle* has an entry in the hard-copy version of LDOCE as:

saddle[2] /'sÆdl/ *v* [T (UP)] to put a saddle on (an animal): *He saddled (up) his horse and rode away.*

This entry provides the word, indicates that it is the second sense of the word (the superscript 2), the pronunciation (/'sÆdl/), the syntactic category (*v*), transitivity information (T) (i.e., that the verb requires an object), optional phrases (UP), the definition and an example sentence. The on-line version has all of this information as well as some extra information. The subject

[3] The words in capital letters in definitions in LDOCE indicate words that are not part of the defining vocabulary but are defined elsewhere in the dictionary.

field code lists this as EQ, an equestrian term. The box code lists a human subject (H) and an animate object (A). These *selectional restrictions* can be of great use to an NLP system by providing clues to the types of thematic roles a noun can support (for instance, inanimate objects cannot be AGENTs) and by limiting the semantic possibilities (to equestrian or figurative uses, for example). This type of use will be explored further in chapter four. Additional information that is available for some entries includes box codes on country of origin, social register, level, period in which the word was used, language of origin, whether it is a new term, and if there are any cross references or illustrations.

Roget's International Thesaurus

Roget's International Thesaurus (RIT) (Roget 1977) is a rich, culturally validated source of information. Roget's original intent for the thesaurus was to provide a "grouping of words according to ideas" and not to produce a list of synonyms. This intent carries into current editions. RIT partitions the world of ideas into eight classes: *abstract relations, space, physics, matter, sensation, intellect, volition* and *affections* (Figure 2.2).

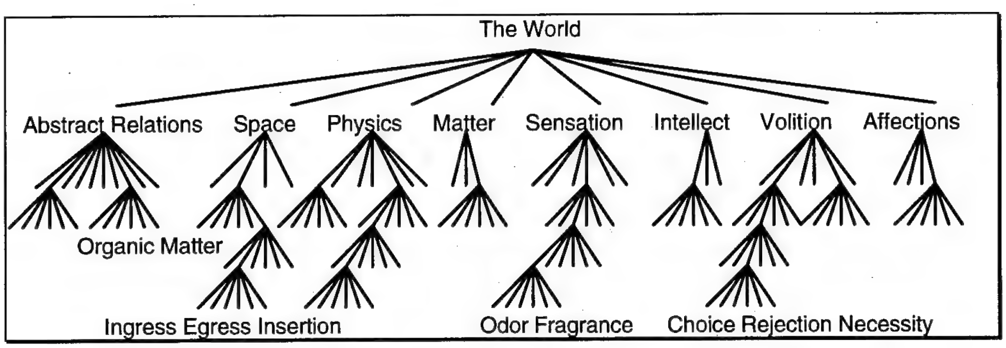

Figure 2.2. The Relational Hierarchy in RIT

Each class is further divided into anywhere from 3 to 10 subclasses. For instance, class five, *sensation*, is divided into 6 subclasses: *sensation in general* and the five senses (*touch, taste, smell, sight, hearing*). Each subclass

is divided into from 0 to 14 headings (ex., *touch* has zero, *hearing* has five). Each heading is divided directly into categories, the traditional entry point into the thesaurus. Each category is a basic semantic concept such as *odor, fragrance* or *stench*. Categories are divided into paragraphs that are grouped mainly by part of speech. Paragraphs are divided into semi-colon groups that contain the individual words. There are 8 classes, 41 subclasses, 183 headings, 1042 categories and approximately 225,000 words in the fourth edition.

The on-line version of RIT (Sedelow 1986) is a mathematical model of the index with some added enhancements. Since it contains just the index, the on-line version is exactly half of the printed version. The printed version contains both the alphabetical index and the words arranged by category number (that is, hierarchically). Either ordering contains sufficient information to recreate the other ordering. For instance, to recreate the hierarchical ordering from the on-line index, one simply has to sort the thesaurus on the field containing the category number[4] . Unlike the hardcopy edition, the on-line version is not arranged in hierarchical order. This limits its capabilities for some uses, such as browsing, as related words are not juxtaposed. The mathematical model makes it more useful in other ways though, as it has been optimized for computational processing by converting much of the structural information to ordinal values.

The lexical size of RIT makes it attractive as an NLP lexical source but even more attractive than mere size is the partitioning of the semantic space. By partitioning the world into classes, subclasses, headings and categories, Roget has created a useful model of the cognitive world. Whether that model is "correct" or cognitively valid is not a concern here[5] . It is assumed that the model is as correct as any other. Our world is constantly changing as are our viewpoints of the world. While no static model can hope to be valid for more than a few points in time, Roget's model has been used by English speaking peoples for 140 years and therefore the model may have become part of our

---

[4] Though since the on-line index is 12 megabytes, sorting it can be a problem.

[5] There is some evidence that our cognitive representation is in fact hierarchical (Miller 1991) though there are probably separate hierarchies for nouns and verbs.

linguistic culture. If that is so, then Roget's model may be more valid than many other semantic models. The point is not really a concern in this research as the emphasis here is not on cognitive validity but rather it is on usefulness for computational processing, and this will be determined empirically.

Summary

This chapter covered the lexical requirements of a PBP and examined those requirements in respect to MRDs. The design of the lexicon itself was then covered. By selecting a trie representation we gain the ability to use very large lexical resources and still have them optimally accessible for parsing systems. The tradeoffs with this type of representation were briefly discussed.

The chapter then provided background on the two lexical resources being used. *LDOCE* was chosen for this research for a number of reasons not the least of which is the richness of lexical information available in the on-line version. Some of the information that is required by the system, and described in chapter four, is not present in other MRDs. For instance, selectional restrictions are explicitly available as box codes in LDOCE but are unavailable in *Webster's Seventh New Collegiate Dictionary*. Other reasons for choosing LDOCE are less theoretical but nonetheless important. These include availability, researcher familiarity and the availability of support from other researchers in the NLP community.

RIT was chosen for very similar reasons. Certainly availability, researcher familiarity and the ability to leverage off other NLP research played a large role in the selection of this resource. More importantly though, is RITs relational hierarchy and the inclusion of related, though not necessarily synonymous, terms. Other computational "thesauri" generally lack a relational hierarchy. They are, in effect, flat lists of synonyms and antonyms. The relational hierarchy makes RIT fairly unique in that respect. Additionally, RIT is much larger than LDOCE so that it is highly unlikely that many words found in LDOCE will be missing from RIT.

Chapter 3
Extracting Extrinsic Knowledge

The course of every intellectual...ends in the obvious, from which the non-intellectuals have never stirred.

A. Huxley

Introduction

This chapter starts with a discussion of some of the problems of going from a general ASCII based dictionary to an MRD. It then covers methods of extracting information from MRDs to create a usable on-line lexicon. Finally we cover the extraction of the extrinsic information found in our MRD version of LDOCE that is required by a PBP.

ASCII to MRD

While it might be possible to read information directly from a pure ASCII file and use it as an MRD, it is easier to reformat the dictionary into a more readily readable state. This is especially true in the case of LDOCE where the original format of the dictionary was formatted for printing (i.e., a printer's tape).

ASCII Codes	Meaning
[5-49-7-5-16]	headword
[5-52-7]	variation
[5-53-7]	syntactic category
[5-54-7]	morphology
[5-57-7-5-11]	subject field code
[5-65-7]	definition
[5-68-7]	example
[5-73-7]	etymology
[5-74-7]	reference
[5-82-7]	grammar code
[5-88-7]	box codes
[5-89-7]	secondary grammar code
[5-91-7]	idiomatic headword
[5-99-7]	idiomatic use
[5-100-7]	idiomatic example

Table 3.1 ASCII Codes and Corresponding Meanings

To start the conversion we identified the control sequences used for printing and converted them into readable English. Some of the codes with their corresponding meanings are given in Table 3.1. Once the files[6] were into this format it was possible to write a program to convert the files into a more readable form. The work at this point was greatly aided by using Guo's dissertation (Guo 1989). The headwords were subsequently identified and each entry under a headword was labeled with: the headword, a sequential index for the headword, the type of entry (ex., definition, syntactic category) and the data for the entry. For instance, an entry for the definition of *pace* would be

(pace,14,definition,"to measure by taking steps of an equal and known length").

This structure is somewhat redundant in that the headword is repeated for each entry. The structure allows us, however, to quickly access the information using UNIX™ facilities such as grep, AWK or a higher level language such as prolog. For example, to find all the definitions that contain the word *measure,* the grep command would simply be

grep ,definition, d? I grep measure

Since the headwords are repeated on every line this would quickly identify the entries for (absolute, acreage, aeon, amp, ..., year).

Similarly if all of the definitions from the file *da* were required to be collected into a separate file, this could be done with AWK by checking if the third field is *definition* and if so then printing the fourth field to a file.

awk -F, '$3 == definition {print $4}' da > defs

This representation for the MRD is extremely handy. It allows the data to be manipulated and thus processed in an extremely efficient manner. Of course, the lexicon does not use this representation but rather the trie representation covered in the last chapter. That means that while the development of the MRD occurs in one form, the ultimate product (the lexicon) is in another, with a corresponding duplication of data and loss of disk space.

---

[6] Since the dictionary was too large to handle as a single file, we broke it into 26 files, one for each letter. These were named da, db, dc, ... dz.

Using On-Line Dictionaries

In his dissertation, Ahlswede (1988) investigated the use of two different approaches in the analysis of dictionary definitions in *Webster's Seventh New Collegiate Dictionary* (W7). The first was an NLP approach that used Sager's Linguistic String Parser (Sager 1981) to completely parse the definitions. The second used the UNIX text processing utilities (ex., grep, sed, lex) to extract patterns from the definitions that were then interactively processed. Both approaches resulted in the development of relational triples of the type *love* SYNONYM *like* or *car* HAS-PART *wheel*. The results were rather mixed.

The NLP component was labor intensive both from a human and machine viewpoint. The component took

> "... a man-year or so of development time for the definition grammar, during which considerable computer time was spent parsing batches of definitions and considerable human time spent poring over bad or failed parses, and rewriting the grammar." (Ahlswede 1988:151).

The considerable computer time turned out to be 180 hours of CPU time on a VAX 8300 to parse the 8,000 or so definitions. The average time per parse varied considerably depending on the syntactic category of the defined word. Adjective phrases took an average of 10.59 seconds per parse while transitive verbs took 48.33 seconds per parse. The parser had an overall success rate of around 70%. The reasons for the parser failing to parse a definition were extremely varied and Ahlswede felt that the minimal improvements expected from rewriting the grammar to improve the success rate would not be worth the amount of effort required.

In contrast to this, by using the text processing utilities approach he was able to generate 11,596 relational triples for the intransitive verb definitions in just three hours. The quality of the triples thus produced was comparable to those produced by parsing.

Viewed in this light, the parsing seems to be wasted effort but Ahlswede does not think so. There seems to be two reasons for this. The first is that the parser that he used is quite slow by current standards. This is due in part to its being a rule based grammar that attempts a comprehensive coverage of English. There are numerous ways of speeding up the parser for this particular application. One of the more obvious ways would be to limit the grammar to only the part needed to cover the linguistic variations found in the dictionary. This could be accomplished by selecting a random subset of definitions and parsing them, keeping track of which rules of the grammar were used. This approach, in essence, would be comparable to writing a grammar specifically for the dictionary but would be less work. Of course, speed in this case would not be optimal because of the considerable overhead associated with the sophisticated user interface of the parser. The second reason that Ahlswede thinks that parsing is worthwhile is of more theoretical importance. The parser was able to identify some relations that were not representable using the relation-triple grouping. For example, the definition of *dodecahedron* is *a solid having 12 plane faces*. The parser gives a syntactic representation for the complete definition but in triples the representation is limited to *dodecahedron IS-A solid*, or *dodecahedron HAS-ATTRIBUTE faces*. It cannot represent *dodecahedron HAS-ATTRIBUTE faces NUMBER 12*. Certainly *12* is not an attribute of *faces* but rather of *dodecahedron*, and then only when referring to the number of faces. This type of relation remains a problem for the simple relational-triple.

Ahlswede's dissertation presents many useful techniques and illuminates many interesting facets of machine-readable dictionaries that are of direct benefit to this research. In particular the analysis of W7, which lacks any explicit selectional restriction information, reinforced the selection of LDOCE for the current research (the information on selectional restrictions should be useful for the selection of thematic roles). The speed of parsing is only of minor concern to the present research and then only in terms of relative speed with different lexicons. The relational-triple representation is not being used for the present work.

Summary

This chapter has covered some of the problems of going from a general ASCII based dictionary to an MRD. It then discussed methods of extracting information from MRDs to create a usable on-line lexicon. Obviously the best representations are useful for domain independent processing only when they contain enough general information to cross domains. That type of information is difficult to acquire in hand-coded lexicons, so the chapter continued with an examination of Ahlswede's work on extracting information from W7.

Chapter 4
Extracting Intrinsic Knowledge

Thirty spokes share the wheel's hub;
It is the center hole that makes it useful.
Shape clay into a cup;
It is the space within
that makes it useful.
Cut doors and windows for a room;
It is their emptiness
that makes them useful.
Therefore profit from what is there;
Utilize what is not.

Lao Tzu, Tao Te Ching

Introduction

As shown in chapter one, the extension of NLP to include domain independent semantics can prove useful to many fields both from within and outside of Information Science. This research posits one way of providing that extension – a principle based parser (PBP) that uses a semantically enriched lexicon automatically derived from machine-readable lexical sources. This chapter covers the extraction, from the MRD, of the basic semantic component of the lexicon — the extraction of the thematic roles.

Thematic Roles

Thematic roles (also called θ-roles) are the roles that nouns play in a sentence; as stated earlier, "Who does what to whom." One of the principles of the parser, the θ-Criterion, requires that each overt noun in the sentence have such a role. These roles are not explicitly present in LDOCE but can, to some extent, be derived.

Methodology

The methodology used to extract the thematic roles is based on the identification of repeated patterns of words in the definitions. It was noticed that lexical patterns such as, *to cause to* are indicative of an AGENT (Liddy and Lauterbach 1993). It was felt that if enough of these patterns existed and could be identified then they might provide sufficient discriminatory power to enable the extraction of the roles from the MRD. The actual number of roles and their precise nature is a matter of contention in the Government-Binding literature. The reason for this is that, syntactically, it makes no difference what name is assigned to the role of the arguments of a verb (Sells 1985), that is a semantic concern. What is important, syntactically, is that the role is assigned according to the principles. Therefore, there is some latitude in

the selection of roles. We decided to use Cook's Matrix Model (Cook 1989) based on the perception of ease of implementation. The Matrix Model, Figure 4.1, uses just five roles: AGENT, THEME (or OBJECT), EXPERIENTIAL, BENEFACTIVE and LOCATIVE.

Verb Types	Basic	Experiential	Benefactive	Locative
1. State	Ts *be tall* Ts, Ts *be* + N	E, Ts *like* Ts, E *be boring*	B, Ts *have* Ts, B *belong to*	Ts, L *be in* L, Ts *contain*
2. Process	T *die* T, T *become*	E, T *enjoy* T, E *amuse*	B, T *acquire* T, B ...	T, L *move*, iv L, T *leak*
3. Action	A, T *kill* A, T, T *elect*	A, E, T *say* A, T, E *amuse* (agt)	A, B, T *give* A, T, B *blame*	A, T, L *put* A, L, T *fill*

Figure 4.1. Cook's Matrix Model

For each verb in the MRD, a θ-role frame was to be created that contained the type for each argument of that verb. For example, as seen in the figure, *acquire* would have a frame similar to *acquire*[BENEFACTIVE, THEME]; indicating that the person (the BENEFACTIVE) who *acquires* something (the THEME) benefits from that which they acquire. These frames were to be developed in as automatic a fashion as possible by using information from: analyzed definitions; subject field codes; the box codes, which provide information on the type of arguments (ex., human or abstract); the grammar codes, which provide information on the transitivity of a verb and the syntactic category of any extra arguments; and other information available from the dictionary. This information was then to be checked to determine if it is consistently available, as meaningful entries are sometimes sparse in the

dictionary. Subject field codes, for instance, are generic codes around 57% of the time. These sparse entries do not provide consistent enough information to be used as the primary source for the frames.

Algorithm

A first attempt at automating the process involved an examination of verb definitions for patterns. All of the definitions for the verbs with frames given in Cook were extracted from LDOCE. These definitions were then pruned to only those senses that matched Cook's frames. The definitions were then divided into three groups; stative, process and action verbs corresponding to the three rows of the model. Each group was then processed and analyzed to identify all repeated lexical patterns of less than length 10. That is, patterns consisting of ten or more words were not specifically identified. The patterns were then restricted to only those unique to a given class of verb. For instance, only those patterns that uniquely appeared in the stative definitions. These unique patterns were then run against the whole MRD and the results were analyzed.

The patterns were able to extract only 67% of the verb definitions. In other words, a full one-third of the verbs have definitions that contain none of the repeated patterns. Of the third that contained the patterns a full 10% of those contained conflicting patterns. For instance, *to cause to*, as mentioned before, indicates an AGENT and thus an action verb. *To come, go* is indicative of a process verb. Therefore, the patterns identified *draw* as both an action and a process verb because its definition is

draw - *to cause to come, go or move by pulling.*

Since each role must be unique this 10% must be re-examined manually. That means that the automatic extraction can only identify 60% of the verbs at best. This is below what is required to actually rely on this process in a fully automatic manner.

The above algorithm only requires a couple of paragraphs to explain but it is the result of months of research. The major time consumers where the extraction of the lexical patterns and ensuring that these patterns were

unique to one verb type. The extractions were done using UNIX™ text processing features (namely, AWK, grep, sed and shell scripts). The patterns were then checked for uniqueness (using sort, uniq and grep). Some problems occurred with sorting the files. They had to be padded with blanks and resorted in order to ensure the results were truly unique. The need for the resorting only became apparent during the failure analysis of the data. Patterns that were supposedly unique were occurring in multiple places. The resorting solved the problem but the extraction of roles had to be completely re-run for all of the patterns. This amounted to extra weeks of effort.

For now this is the extent of the extraction process. Much work remains to be done but it is no longer within the scope of this effort. The current results indicate that the extraction cannot be done in an automatic manner but would probably benefit from a semi-automatic, tool-type, of approach. Before this result is accepted, however, some further testing should be done.

First, the testing of patterns was only done for the rows. It should be investigated if the columns work better than the rows. If they do, then the results from the columnar extraction may be sufficient to properly discriminate the rows. Second, the testing to date has involved repeated lexical patterns. This testing should probably be expanded to include repeated syntactic patterns. This could be accomplished by parsing the definitions and running the pattern extraction programs on the parses. It is anticipated that the resulting patterns might prove useful in discriminating the proper roles. It is certainly worth testing.

Testing

The result of the above extraction procedures would be θ-role frames for each verb in LDOCE. The identification of unique roles for the frames is not enough however, the correctness of these roles also has to be verified. This can be done in three ways.

Since the example sentences in LDOCE are grammatical, the parser can verify the correctness of form for each θ-role frame by using the frame to parse the corresponding example sentence. Failure to parse a sentence would

indicate an ill-formed frame, which could then be rejected. Such a test would verify the number of roles.

Of equal importance to the number of roles is how many of the extracted roles are semantically correct. The correctness of the roles, for the verbs found in Cook, should be judged by comparing them manually to the roles given there. The rest of the frames could be judged by manually inspecting randomly selected samples.

Additionally, an analysis of the results could be given, addressing such issues as the part that LDOCE's defining vocabulary plays in the extraction process, the consistency of the definition patterns, and the reliability of other sources of information in LDOCE that are used (ex., box codes, subject field codes).

The issue of reliability, especially of the box codes, presents an opportunity for an additional test. It should be possible to construct another filter for the parser using the information in the box codes regarding selectional restrictions of the θ-roles. For instance, if a verb had a θ-role frame of [AGENT, THEME], the newly written semantic filter would check the box code of the subject noun to verify that it is animate and thus capable of being an AGENT. If it was not, then the filter would rule out the frame, and the corresponding structure, as being semantically anomalous. Analyzing the output of running the parser with the new filter on the LDOCE corpus should provide some interesting results. It should reduce the number of parses per sentence and improve the ability to correctly attach arguments (for instance, prepositional phrases). At the very least it would provide a demonstration of using the semantic information supplied by the θ-role to guide the parser in identifying correct structures.

Summary

This chapter has shown that lexical patterns in the definitions of LDOCE can be exploited in extracting thematic roles. While the time constraints of this effort did not allow complete testing of the approach it did show that it is possible to extract the thematic roles from the MRD at least to the point where a semi-automatic tool could be constructed to aid a human user in determining the proper role. While the results are not conclusive they are very encouraging. Further work remains to be done especially in attempting to use syntactic patterns in place of the lexical ones used here.

The next chapter will cover one method of enhancing the semantics for words by mapping the definitions of the word to its proper place in a relational hierarchy.

Chapter 5

Mapping Word Senses into Roget's

We say: the essential thing in a word is its meaning. We can replace the word by another with the same meaning. That fixes a place for the word, and we can substitute one word for another provided we put it in the same place.

L. Wittgenstein

Introduction

This chapter covers our efforts at mapping the word senses, associated with the definitions in LDOCE, to their proper places in the relational hierarchy in RIT (Figure 5.1).

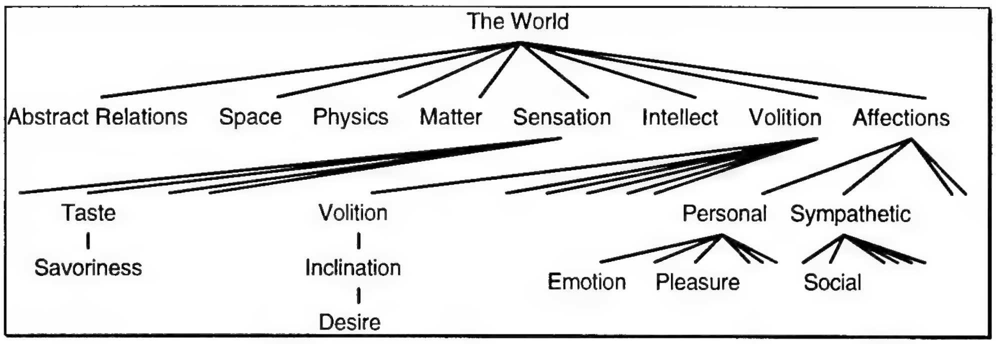

Figure 5.1. The relational hierarchy in RIT

Mapping

The mapping not only labels the word sense with one of the 1042 "semantic primitives" found in RIT, but also places the word into the RIT relational hierarchy at the semi-colon group level.[7] The result is a fairly rich semantic classification of words that is assumed to provide a rich semantics suitable for multiple purposes. The exact nature of the use of this classification is beyond the scope of the current research, but it would allow processing similar to that done in the SCISOR system (Jacobs and Rau 1988, 1990) on a less domain dependent basis.

The difficulty with this mapping process is that it is somewhat ambiguous. Each word is found in the hierarchy in a large number of places. For example, *love* appears in RIT in five different places. A closely related

[7] The inclusion of the semantic primitive label is somewhat arbitrary but is being done to ease the analysis of the output. It is easier to intellectually understand the meaning of the labels than it is to understand the meaning of the semi-colon group number.

word, *want*, occurs eight times in RIT, but in only one of these is the meaning of *want* the same as the meaning of *love*. That is, these two words alone appear in RIT in twelve distinct places only one of which would be the proper mapping for their shared sense.

The process of mapping the words into the hierarchy was based on the view that the thesaurus is a collection of related terms. The closer the semantic relatedness between two words, whatever that relatedness is, the closer the words are in the hierarchy. Therefore measures of distance in the hierarchy can be roughly construed as measures of relatedness or semantic distance. Various methods of measuring the distance were investigated. These methods ranged in complexity from the method of quartets (Talburt and Mooney 1988) to as simple a method as counting the number of intervening words found when using a standard tree traversal algorithm.

Each sense of each word in LDOCE could be placed in the RIT hierarchy using one of a number of methods. It is observed, though, that a definition of a word defines the semantic space or environment for that word. That is, the definition must recreate enough of the sense of the word that a user can properly relate the defined word to words that are already known. If distance in the hierarchy is considered as a measure of semantic relatedness, then each word used in the definition could be measured to see how close it is to the different spaces in the hierarchy where the defined word is found. The closer the semantic distance, the better the fit. For ease of understanding, an example will be given.

Example

Measurement of relatedness (MOR), in this example, will be normalized between zero and one and will be based mainly on the depth of the hierarchy. The maximum depth is 7. That is, there are seven levels in the hierarchy: 0, the world; 1, classes; 2, sub-classes; 3, headings; 4, categories; 5, sub-categories; and 6, semi-colon groups. If two words are in the same semi-colon group they will be said to have a MOR of nearly 1 (i.e., highly related),

if they are in different classes then they will have a MOR of nearly zero. MOR will be computed using the formula **MOR** = (**L** / 7) + (1 / (**S** * 7)), where **L** is the lowest level of the hierarchy that the words share and **S** is the number of subdivisions in the level **L,** and 7 is the total depth of the hierarchy. For instance, the words *rain* and *stream* are under the same heading (Class Four: *Matter*, SubClass: *Inorganic Matter*, Heading: *Liquids*) so **L** for these words would be 3. Since there are 13 categories under the heading *Liquids* (ex., *Liquefaction, Moisture, Ocean*), **S** would be 13. The measurement of relatedness between *rain* and *stream* would therefore be **MOR**=3/7+(1/(13*7))=0.4394. The actual value that the formula returns is not as important as the relative values among the words of a given definition. This particular formula is probably a good first approximation. The first term (**L**/7), reflects the level that the two words share. The value of **L** increases when the words share more levels. Thus, the value of **MOR** increases the closer the relationship between the words. The second term (1/(**S***7)), reflects the number of branches in the hierarchy at the level where the words differ. The number of branches reflects the amount of fine-grained distinctions in the partitioning of semantic space.

feel	discrimination 492.1, grope 485.5, knack 733.6, milieu 233.3, texture 351.1, touch 425.1, appear to be 446.10, emotions 855.11, experience 151.8, intuit 481.4, sense 422.8, suppose 499.10, touch 425.6
love	be fond of 931.18, desire 634.14, enjoy 865.10, savor 428.5, have deep feelings 855.14
desire	hope 888.1, intention 653.1, love 931.1, request 774.1, sexual desire 419.5, thing desired 634.11, will 621.1, wish 634, be eager 635.7, be hopeful 888.7, intend 653.4, lust 419.22, request 774.9, will 621.2, wish 634.14
strong	accented 594.31, alcoholic 996.36, eloquent 600.11, energetic 161.12, forceful 159.13, great 34.6, influential 172.13, malodorous 437.5, powerful 157.12, pungent 433.8, rancid 692.42, robust 685.10, strong-smelling 435.10, substantial 3.7, tough 359.4
friendship	927

Figure 5.2 Entries in RIT

The first definition of the verb *love* is "to feel love, desire, or strong friendship (for)". By comparing the distances of the main words in the definition (i.e., ignoring prepositions, determiners and the like) to the five locations in RIT, it would be discovered that the best fit occurs with the words at 634-*Desire*. The entries in RIT for the five main words[8] are given in Figure 5.2.

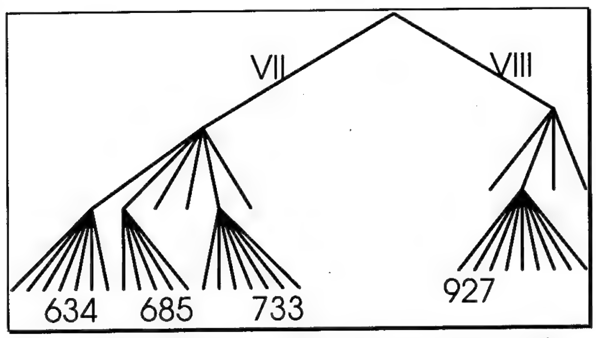

Figure 5.3 Hierarchy containing *Love*

Figure 5.3 illustrates the part of the hierarchy that is under consideration here. The closest occurrence of *feel* to 634, is at 733.6. That is in the same class but in a different subclass than 634; **L**=1, **S**=6, **MOR**=0.1666. *Love* and *desire* are both at 634; **MOR**=0.9996. The nearest occurrence of *strong* is at 685; **L**=1, **S**=6, **MOR**=0.1666. *Friendship* only occurs at 927 (in Class Eight: *Affections*); **L**=0, **S**=8, **MOR**=0.0178. That gives a total sum of 2.3502 for the five words. These values are given in the second column of Figure 5.4. The figure also gives the measurements for the other four senses of *love*. Notice that 2.3502 is the maximum of the totals and therefore 634 is the closest fit. In a similar manner, the second definition of *love*, "to have a strong liking for; take pleasure in" would best fit with the words at 865-*Pleasure*. Thus the senses of LDOCE could be matched to the senses of RIT by computing average relatedness values between the words in a definition of a candidate word from LDOCE with each category of RIT in which the candidate occurs.

[8] The values of **S** cannot be deduced from the figure. They have to be calculated directly from the thesaurus.

	428.5	634.14	855.14	865.10	931.18
feel	0.1666	0.1666	0.4333	0.3014	0.1785
love	0.9996	0.9996	0.9996	0.9996	0.9996
desire	0.0178	0.9996	0.3015	0.3014	0.1785
strong	0.4462	0.1666	0.1785	0.1785	0.4443
friendship	0.0178	0.0178	0.1785	0.1785	0.4443
Totals	1.6480	2.3502	2.0914	1.9594	1.9794

Figure 5.4. Relatedness measures to *Love*

The exact details of the algorithm need refinement. For instance, the above algorithm does not take into account the order of the hierarchy or its entries only the number of branches at any given level. The order is not random and should be considered. It was determined empirically that the order does produce superior results to the above algorithm. Therefore, the formula should be changed appropriately. The actual formula that we use to do the mapping will not be presented here as it is patent pending.

Testing

The testing of the algorithms was done using the following process. First, a sample of 100 word senses was randomly selected from LDOCE. These senses were then manually mapped to the RIT hierarchy in the place(s) that was felt was the most appropriate. This manual mapping became the baseline and was thereafter considered as the "correct" mapping.

The algorithms were then run against the baseline and the percentages of correctly mapped senses were computed. This process was run on a number of different algorithms including the method of quartets (Talburt and Mooney), the difference in category numbers, and a standard tree traversal. The end result (the algorithm that is patent pending) is a combination of approaches.

Results

The best of the algorithms could correctly map the word senses from LDOCE to the RIT hierarchy about 63% of the time. While this is not as high

as we would have liked, it is high enough to provide the basis of a semi-automatic tool. The tool could provide candidate locations along with the evidence it has compiled for each location. If we pursue the problem, this is probably the direction in which we will go.

Multi-media Lexical Browser

As a by-product, more or less, of the research we created a multi-media (hypertext) demonstration of an integrated RIT/LDOCE lexical browser for aero-space terminology.

Our browser is an attempt at using multi-media technology to provide users with an interesting way of exploring the possibilities presented by tightly coupled lexical resources. At top level the user has the choice of using either an alphabetical listing of words or the portion of the RIT hierarchy shown in Figure 5.5.

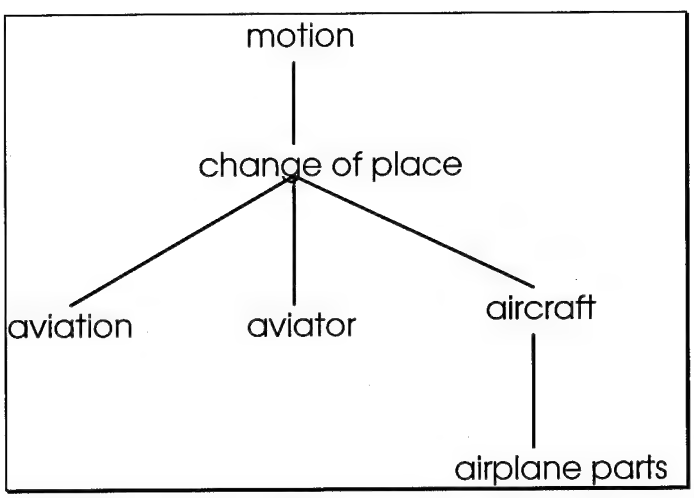

Figure 5.5 Aerospace-Terminology Sub Hierarchy

The top two nodes of the hierarchy, *motion* and *change of place*, are not used as links because the hierarchy is not complete. It is restricted to only aerospace terminology. The bottom four nodes are links and clicking on one of these produces a lexis (list of words) of related terms (ex., aviation terms).

Clicking on one of the related words produces the definition for that word. Each definition includes links to any aerospace terms that are used in the definition. There are also buttons that relate to hyper-graphics, the current lexis, or speech (Figure 5.6).

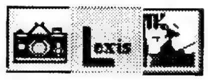

Figure 5.6 Buttons

The hypergraphics include pictures of airplanes and helicopters. Parts of these pictures are themselves links, thus the term hypergraphic. The lexes were covered above. The speech output includes the word and a sample sentence produced by using the pronunciation and example sentences from LDOCE.

The system was written as a Windows™ 3.1 help file for the PC. This approach was chosen for a number of reasons: Windows provides a common environment with which many users are already familiar. There are a large number of tools available that ease the production of help files and hypergraphics. The Windows environment makes the interconnection of text, graphics and speech easier. The actual file can be written using any word processor or editor that can save in rich text format (rtf), and then compiled. In our case we used Microsoft Word as the word processor, saved the files in rtf, added the hypergraphics using shed.exe that is in the Windows Software Development Kit, which is where we also got the help compiler. The text to speech component was part of the Soundblaster[9] utilities.

The end result is a rather nice demonstration of some of the functionalities that are possible with tightly coupled lexical resources. The most obvious use of such a tool would be for people that need to explore a new domain in depth, in this capacity the browser would be an aid to learning.

---

[9] Windows and Word are trademarks of Microsoft Corporation. Soundblaster is a trademark of Creative Labs, Inc.

Summary

This chapter described the research that was accomplished in determining the extent that words, as defined in on-line dictionaries, can be automatically placed in their appropriate location in the semantic hierarchy found in Roget's International Thesaurus. A question that often arises concerning this research is, "Why bother?"

Our original motivation in doing the mapping was to supply a readily available, computable form of semantics to the words in LDOCE. Obviously, the definitions themselves are rich semantic representations, but they lack the quality of being readily computable. To understand the semantics of a definition requires an understanding of all the included words and an understanding of how those words interrelate within the definition. The hierarchy has none of this. It simply supplies a relative measure of relatedness for the words. That is, it can indicate which words are more closely related without indicating how they are related.

As the work with the mapping progressed, a larger use for it was realized. By utilizing the hierarchy as a common semantic representation, disparate lexicons can be integrated. If the integration can be done well then lexicons from existing NLP systems designed for small subfields of C^3I could be integrated to form a comprehensive C^3I NLP system. This would not only substantially reduce development time but increase coverage and portability.

The next chapter will give an overview of the entire research and the significance of the results.

Chapter 6
Summary

Command and Control (C2) is critically dependent upon the timely provision of appropriate information. One reason the role of military commanders is difficult is the lack of information. "How many entities are there? Where are they? What are their movements? Which are hostile? Are they a threat to me? What resources are available to me? How can I most effectively deploy my resources? What are my chances of success?"

D. Whitaker, Defence Research Agency

Communication is the transfer of information from one person to another. While a picture may be worth a thousand words (though we are never told which thousand) most communication remains language based. For applications like C^3I, the amount of language based information is enormous and growing. Any realistic software system that is intended to positively impact the C^3I domain then must be capable of handling this language based information. Natural Language Processing (NLP) is the primary field concerned with the creation of a system that understands such unconstrained language based information (i.e., text and speech). As such, NLP must be considered as an integral component of future C^3I systems.

While current NLP systems can do in-depth processing of text only in constrained domains, linguistic based NLP systems provide a strong theoretical base for expanding our capabilities to unconstrained domains. This research has posited one approach that may be used to accomplish that expansion.

Our original intent was to demonstrate an approach toward domain independent semantic processing: the combination of a robust, domain independent syntactic system with a large, general lexicon that had been tightly linked with a relational hierarchy. Various constraints made that demonstration impossible.

What was shown was that a machine readable dictionary (MRD) provides sufficient lexical power to be used as a lexicon for a domain independent parser. Not only does the MRD provide sufficient information, but does so far easier and at less cost than creating a comparable lexicon by hand. Further, it was shown that the lexicon thus derived can be tightly linked with a relational hierarchy providing a more readily computable semantics than would otherwise be possible. The added benefit of the linked dictionary/thesaurus is the ability to integrate disparate lexicons thus providing further benefits in portability and integration of existing NLP systems.

While the early stages of this research should not be taken as being conclusive, they do indicate that the problem cannot be solved automatically but probably could be approached semi-automatically. The tools should be built, the lexicon should be developed, the system should be tested, and the research should be continued.

Bibliography

Ahlswede, T.E. (1988) *Syntactic and Semantic Analysis of Definitions in a Machine-Readable Dictionary*, Ph.D. Dissertation, Illinois Institute of Technology.

Berwick, R. C. and S. Fong (1990). "Principle Based Parsing: Natural Language Processing for the 1990s". In P.H. Winston and S.A. Shellard (Eds.), *Artificial Intelligence at MIT: Expanding Frontiers.* pp. 286 - 325. The MIT Press: Cambridge, MA.

Boguraev, B. and T. Briscoe (1989) *Computational Lexicography for Natural Language Processing*, Boguraev, B. and T. Briscoe (eds.), Longman: New York.

Chomsky, N. (1986) *Knowledge of Language: Its Origins, Nature, and Use,* Praeger Publishers, NY.

______ (1986) *Barriers*. MIT Press: Cambridge, MA.

Cook, W. (1989) *Case Grammar Theory*. Georgetown University Press: Washington.

Dorr, B. (1987) *UNITRAN: A Principle-Based Approach to Machine Translation*, AI Technical Report 1000, Master of Science, Department of Electrical Engineering and Computer Science, MIT: Cambridge, MA.

Fredkin, E. (1960) "Trie Memory". *CACM* 3(9):490-499.

Guo, C-M. (1989) *Constructing a Machine-Readable Dictionary from "Longman Dictionary of Contemporary English"*, Ph.D. Dissertation, New Mexico State University: Las Cruces, NM.

Jacobs, P. S. and L. F. Rau (1990) "SCISOR: Extracting Information from On-line News," *Communications of the ACM*, Vol. 33, No. 11, November 1990.

Lau Tzu (550 B.C.) *Tao Te Ching*, trans. Gia-Fu Feng and J. English, Vintage Books: New York, 1972.

Liddy, E. D. and D. M. Lauterbach (1993) "Automatic Construction of a Semantic Lexicon for use in Natural Language Processing Systems". *Final Report for Research Initiation Program*, Rome Laboratory, Griffiss AFB, NY.

LDOCE (1987) *Longman Dictionary of Contemporary English.* Longman: Harlow, UK.

LDOCE (1981) *Longman Lexicon of Contemporary English.* Longman: Harlow, UK.

Miller, G.A. (1991) *The Science of Words*, Scientific American Library: New York.

Morris, J. and G. Hirst (1991) "Lexical Cohesion by Thesaural Relations as an Indicator of the Structure of Text," *Computational Linguistics*, 17(1), 21-48.

Roget, P.M. (1852) *Thesaurus of English Words and Phrases, Classified and Arranged so as to Facilitate the Expression of Ideas and Assist in Literary Composition.*

Roget (1977) *Roget's International Thesaurus, Fourth Edition.* R.L. Chapman (ed.), Harper & Row: New York.

Roget (1992) *Roget's International Thesaurus, Fifth Edition.* R.L. Chapman (ed.), Harper Collins: New York.

Sedelow, S. and W. Sedelow (1986) "Thesaural knowledge representation," In *Proceedings, 2nd Annual Conference of the University of Waterloo Centre for the New Oxford English Dictionary: Advances in Lexicology,* University of Waterloo: Waterloo, Ontario.

Talburt, J.R. and D.M. Mooney (1989) "Determination of Strongly-Connected Components in Abstract Thesauri by the Method of Quartets," In *Proceedings of the Workshop in Applied Computing '89,* Oklahoma State University: Stillwater, OK.

Wittgenstein, L. (1934) *Philosophical Grammar.* trans. A. Kenney, University of California Press, Berkeley, CA. 1978.

☆U.S. GOVERNMENT PRINTING OFFICE: 1994-610-126-50099

MISSION

OF

ROME LABORATORY

Mission. The mission of Rome Laboratory is to advance the science and technologies of command, control, communications and intelligence and to transition them into systems to meet customer needs. To achieve this, Rome Lab:

a. Conducts vigorous research, development and test programs in all applicable technologies;

b. Transitions technology to current and future systems to improve operational capability, readiness, and supportability;

c. Provides a full range of technical support to Air Force Materiel Command product centers and other Air Force organizations;

d. Promotes transfer of technology to the private sector;

e. Maintains leading edge technological expertise in the areas of surveillance, communications, command and control, intelligence, reliability science, electro-magnetic technology, photonics, signal processing, and computational science.

The thrust areas of technical competence include: Surveillance, Communications, Command and Control, Intelligence, Signal Processing, Computer Science and Technology, Electromagnetic Technology, Photonics and Reliability Sciences.